# Distributed Computing

## Homework 3: Black Board

Thomas Weise · 汤卫思

tweise@hfuu.edu.cn · http://www.it-weise.de

Hefei University, South Campus 2
Faculty of Computer Science and Technology
Institute of Applied Optimization
230601 Shushan District, Hefei, Anhui, China
Econ. & Tech. Devel. Zone, Jinxiu Dadao 99

合肥学院 南艳湖校区/南2区
计算机科学与技术系
应用优化研究所
中国 安徽省 合肥市 蜀山区 230601
经济技术开发区 锦绣大道99号

# Outline

1. Task

website

# Overview

- Install the GlassFish servlet container/application server (or any alternative) [1–3]
- Implement a simple Black Board backed by a JavaServer Page [4–8]
- Style the Black Board with CSS
- Deploy this implementation

# Task

- Create a dynamic web application in form of JavaServer Pages

# Task

- Create a dynamic web application in form of JavaServer Pages
- This application should have

# Task

- Create a dynamic web application in form of JavaServer Pages
- This application should have:
    - A web page showing a list of entries, similar to a guest book or black board

# Task

- Create a dynamic web application in form of JavaServer Pages
- This application should have:
  - A web page showing a list of entries, similar to a guest book or black board
  - A `<form>` where someone can write a text into a `<textarea>` and click a submit button

# Task

- Create a dynamic web application in form of JavaServer Pages
- This application should have:
  - A web page showing a list of entries, similar to a guest book or black board
  - A `<form>` where someone can write a text into a `<textarea>` and click a submit button
  - The text should then become the newest entry in the black board and displayed as a paragraph ( `<p>...text...</p>` )

# Task

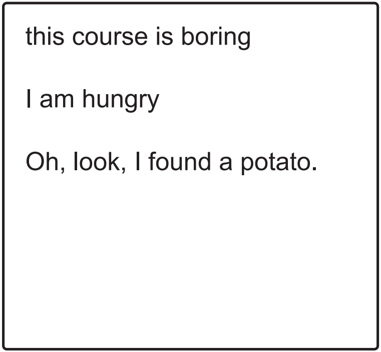
- Create a dynamic web application in form of JavaServer Pages
- This application should have:
    - A web page showing a list of entries, similar to a guest book or black board
    - A `<form>` where someone can write a text into a `<textarea>` and click a submit button
    - The text should then become the newest entry in the black board and displayed as a paragraph ( `<p>...text...</p>` )
    - Cascading Style Sheets (CSS) are used to set the font, text color, and background color of the black board, post texts, and text areas

# Task

- Create a dynamic web application in form of JavaServer Pages
- This application should have:
    - A web page showing a list of entries, similar to a guest book or black board
    - A `<form>` where someone can write a text into a `<textarea>` and click a submit button
    - The text should then become the newest entry in the black board and displayed as a paragraph ( `<p>...text...</p>` )
    - Cascading Style Sheets (CSS) are used to set the font, text color, and background color of the black board, post texts, and text areas
- No user management is required, i.e., you do not need to make a log-in screen

# Task

- Create a dynamic web application in form of JavaServer Pages
- This application should have:
  - A web page showing a list of entries, similar to a guest book or black board
  - A `<form>` where someone can write a text into a `<textarea>` and click a submit button
  - The text should then become the newest entry in the black board and displayed as a paragraph ( `<p>...text...</p>` )
  - Cascading Style Sheets (CSS) are used to set the font, text color, and background color of the black board, post texts, and text areas
- No user management is required, i.e., you do not need to make a log-in screen
- Submit to me: all source codes and files, including your `.war` , in an archive named `hw03_[your_student_id].zip` (where `[your_student_id]` is replaced with your student id)

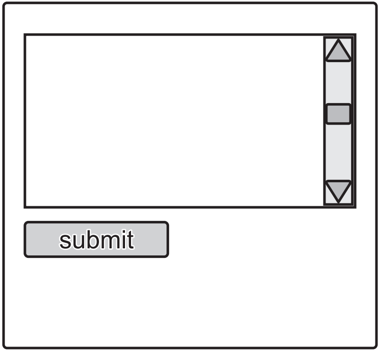


this course is boring

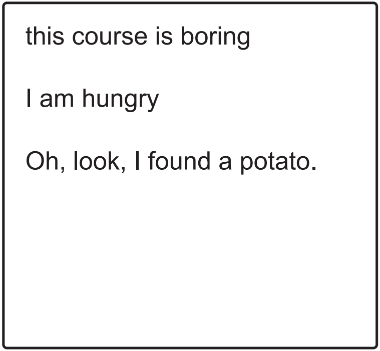
I am hungry

Oh, look, I found a potato.

blackBoard.jsp



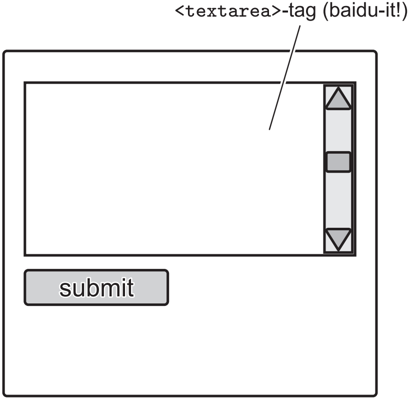
addEntry.jsp


this course is boring

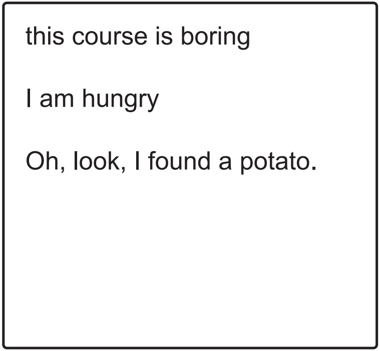
I am hungry

Oh, look, I found a potato.

blackBoard.jsp



addEntry.jsp



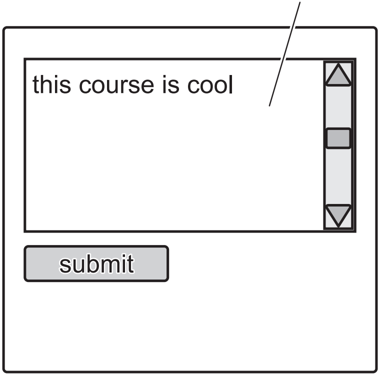
this course is boring

I am hungry

Oh, look, I found a potato.

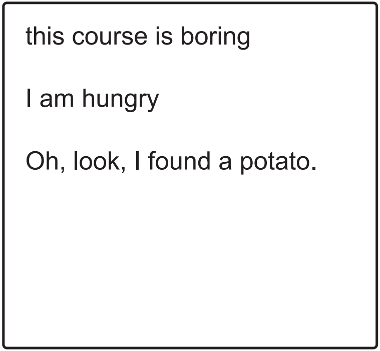
blackBoard.jsp

<textarea>-tag (baidu-it!)



this course is cool

addEntry.jsp

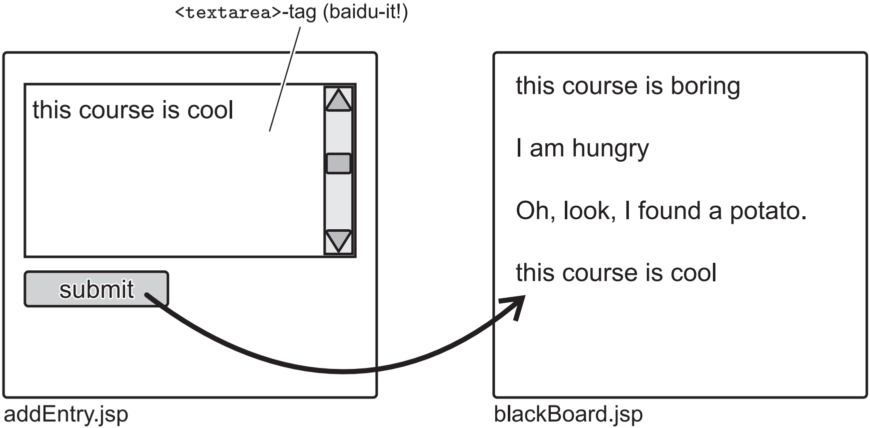


this course is boring

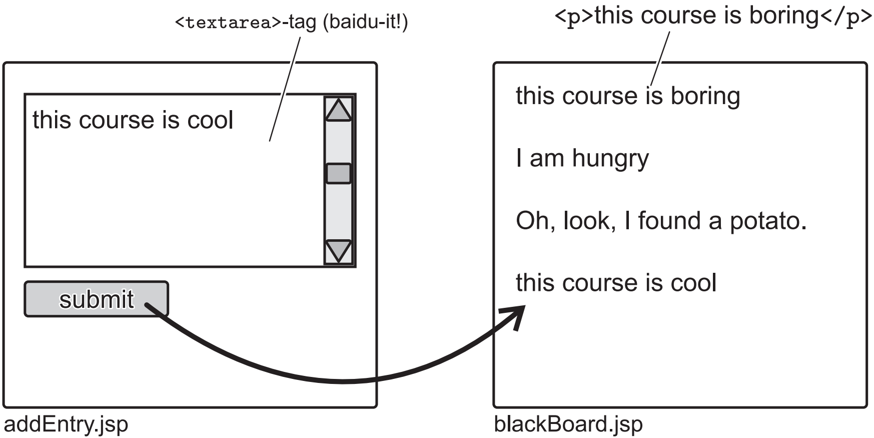
I am hungry

Oh, look, I found a potato.

blackBoard.jsp



<textarea>-tag (baidu-it!)
this course is cool
submit
addEntry.jsp
this course is boring
I am hungry
Oh, look, I found a potato.
this course is cool
blackBoard.jsp



<textarea>-tag (baidu-it!)
this course is cool
submit
addEntry.jsp
<p>this course is boring</p>
this course is boring
I am hungry
Oh, look, I found a potato.
this course is cool
blackBoard.jsp

# Tipps

- This task is somewhat similar to the *shopping cart* example in the lecture

# Tipps

- This task is somewhat similar to the *shopping cart* example in the lecture:
    - The entries of the black board can be stored in a Java Bean that manages a list

# Tipps

- This task is somewhat similar to the *shopping cart* example in the lecture:
    - The entries of the black board can be stored in a Java Bean that manages a list
    - Different from the shopping cart example, the bean must have *application* scope

# Tipps

- This task is somewhat similar to the *shopping cart* example in the lecture:
  - The entries of the black board can be stored in a Java Bean that manages a list
  - Different from the shopping cart example, the bean must have *application* scope if it had session scope, people could only read their own entries. . .

# Tipps

- This task is somewhat similar to the *shopping cart* example in the lecture:
  - The entries of the black board can be stored in a Java Bean that manages a list
  - Different from the shopping cart example, the bean must have *application* scope if it had session scope, people could only read their own entries...
- The task is also similar to the *voter* example in the lecture

# Tipps

- This task is somewhat similar to the *shopping cart* example in the lecture:
  - The entries of the black board can be stored in a Java Bean that manages a list
  - Different from the shopping cart example, the bean must have *application* scope if it had session scope, people could only read their own entries…
- The task is also similar to the *voter* example in the lecture:
  - instead of using a bean, you could also maintain the whole black board text (including the `<p>` tags) as a single global `String` variable

# Tipps

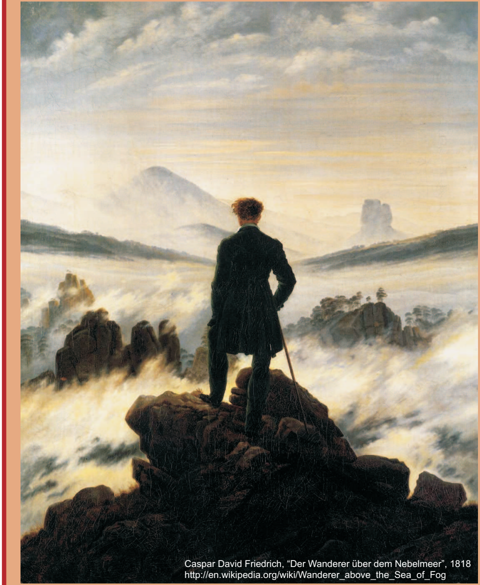
- This task is somewhat similar to the *shopping cart* example in the lecture:
    - The entries of the black board can be stored in a Java Bean that manages a list
    - Different from the shopping cart example, the bean must have *application* scope if it had session scope, people could only read their own entries...
- The task is also similar to the *voter* example in the lecture:
    - instead of using a bean, you could also maintain the whole black board text (including the `<p>` tags) as a single global `String` variable
- You can re-use/modify files from the examples (e.g., `web.xml`...)

# 谢谢

# Thank you

Thomas Weise [汤卫思]
tweise@hfuu.edu.cn
http://www.it-weise.de

Hefei University, South Campus 2
Institute of Applied Optimization
Shushan District, Hefei, Anhui,
China



Caspar David Friedrich, “Der Wanderer über dem Nebelmeer”, 1818
http://en.wikipedia.org/wiki/Wanderer_above_the_Sea_of_Fog

# Bibliography

# Bibliography I

1. *Glassfish*. Redwood Shores, CA, USA: Oracle Corporation, revision 20130208.5d7f765 edition, 2013. URL `http://glassfish.java.net/`.
2. David Heffelfinger. *Java EE 6 with GlassFish 3 Application Server*. Birmingham, UK: Packt Publishing Limited, 2010. ISBN 1849510377 and 9781849510370. URL `http://books.google.de/books?id=GCFdQ5SxPnQC`.
3. Antonio Goncalves. *Beginning Java EE 6 with GlassFish 3*. Expert's Voice in Java Technology. New York, NY, USA: Apress, Inc., August 24, 2010. ISBN 143022889X and 9781430228899. URL `http://books.google.de/books?id=8pQbMr5X-yMC`.
4. Pierre Delisle, Jan Luehe, and Mark Roth. *JSR 245: JavaServer Pages™ 2.1 Final Release*, volume 245 of *Java Specification Requests (JSR)*. May 8, 2006. URL `http://jcp.org/aboutJava/communityprocess/final/jsr245/index.html`.
5. Hans Bergsten. *JavaServer Pages*. The Java Series. Upper Saddle River, NJ, USA: Prentice Hall International Inc., Santa Clara, CA, USA: Sun Microsystems Press (SMP), and Reading, MA, USA: Addison-Wesley Professional, 2003. ISBN 0596005636 and 9780596005634. URL `http://books.google.de/books?id=JNE8sqfQNAUC`.
6. Vivek Chopra, Jon Eaves, Rupert Jones, Sing Li, and John T. Bell. *Beginning JavaServer Pages*. Wrox Beginning Guides. New York, NY, USA: John Wiley & Sons Ltd., 2005. ISBN 0764589520 and 9780764589522. URL `http://books.google.de/books?id=du6Y1dROmAUC`.
7. Javaserver pages technology. In *The Java EE 5 Tutorial: For Sun Java System Application Server 9.1*, chapter 5. Redwood Shores, CA, USA: Oracle Corporation, 2010. URL `http://docs.oracle.com/javaee/5/tutorial/doc/bnagx.html`.
8. Mukesh Prasad. Jsp tutorial, November 11, 2012. URL `http://www.jsptut.com/`.